ASSIGNMENT- 1

SHAIK MASTAN VALI - EE18BTECH11039

Download all python codes from

https://github.com/Mastan1301/EE3025/tree/main/Assignment−1−C/codes/

data files from

https://github.com/Mastan1301/EE3025/tree/main/Assignment−1−C/data/

and latex-tikz codes from

https://github.com/Mastan1301/EE3025/tree/main/Assignment−1−C

1 Problem

The command

```
output_signal = signal.lfilter(b,a,
    output_signal)
```

in Problem 2.3 is executed through following difference equation

$$\sum_{m=0}^{M} a(m)\, y(n-m) = \sum_{k=0}^{N} b(k)\, x(n-k) \tag{1.0.1}$$

where input signal is $x(n)$ and output signal is $y(n)$ with intial values all 0. Replace **signal.filtfilt** with your own routine and verify

2 Solution

Using the properties of z-transform

$$\mathcal{Z}\{x(n-k)\} = z^{-k}X(z) \tag{2.0.1}$$

$$\mathcal{Z}\{y(n-m)\} = z^{-m}Y(z) \tag{2.0.2}$$

where $X(z)$ and $Y(z)$ are the respective z-transforms of $x(n)$ and $y(n)$ respectively.
Applying z-transform on both sides in (1.0.1),

$$Y(z)\sum_{m=0}^{M} a(m)\, z^{-m} = X(z)\sum_{k=0}^{N} b(k)\, z^{-k} \tag{2.0.3}$$

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^{N} b(k)\, z^{-k}}{\sum_{m=0}^{M} a(m)\, z^{-m}} \tag{2.0.4}$$

3 Implementation in C

We first generate the *x.dat* file by reading the data from *Sound_Noise.wav* file and storing it in a dat file. Next, we generate *H.dat* file which is the DFT of the filter's impulse response, that is computed using the filter coefficients b, a.
The python code for doing this is-

codes/generateDat.py

Then, we read the *x.dat* file to obtain $x[n]$ and compute the DFT of x. Then we obtain the output of the filter in frequency domain as-

$$Y(K) = H(K)\, X(K) \tag{3.0.1}$$

Then we store Y in *Y.dat* file and the time domain signal (computed using ifft()) y in *y.dat* file.
This is done using the below C program.

codes/fft.c

Compile the program using-

```
gcc fft.c −o fft −lm
```

Run the program using -

```
./fft
```

Finally, we store the outputs in wav files and plot the two outputs, one done using C and the other using the library functions. The python code for this is -

codes/ee18btech11039.py

The soundfile obtained using C program is -

data/Sound_With_ReducedNoise_1.wav

The soundfile obtained using library function -

data/Sound_With_ReducedNoise_2.wav

4 Verification

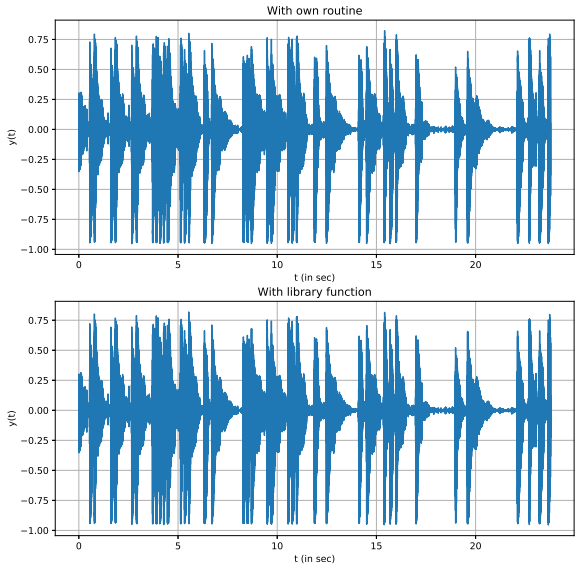

Fig. 0: Time domain response

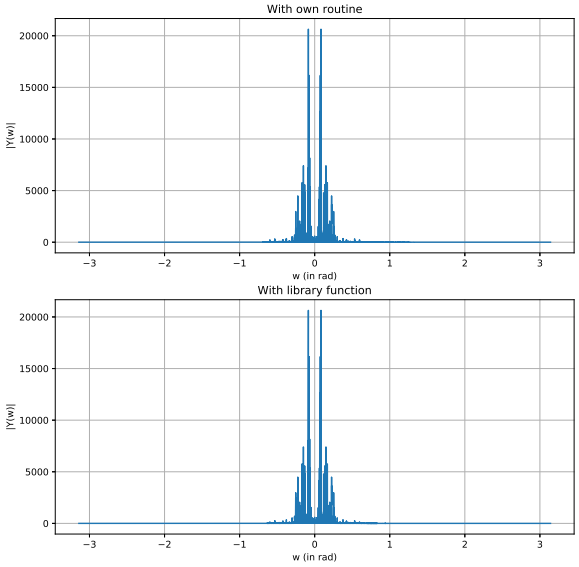

Fig. 0: Frequency domain response